This is an author produced version of *Generating indicative-informative summaries with SumUM* .

Generating Indicative-Informative Summaries with SumUM

Horacio Saggion∗
University of Sheffield

Guy Lapalme†
Université de Montréal

We present and evaluate SumUM, *a text summarization system that takes a raw technical text as input and produces an indicative informative summary. The indicative part of the summary identifies the topics of the document, and the informative part elaborates on some of these topics according to the reader's interest.* SumUM *motivates the topics, describes entities, and defines concepts. It is a first step for exploring the issue of dynamic summarization. This is accomplished through a process of shallow syntactic and semantic analysis, concept identification, and text regeneration. Our method was developed through the study of a corpus of abstracts written by professional abstractors. Relying on human judgment, we have evaluated indicativeness, informativeness, and text acceptability of the automatic summaries. The results thus far indicate good performance when compared with other summarization technologies.*

1. Introduction

A summary is a condensed version of a source document having a recognizable genre and a very specific purpose: to give the reader an exact and concise idea of the contents of the source. In most cases, summaries are written by humans, but nowadays, the overwhelming quantity of information,[1] and the need to access the essential content of documents accurately in order to satisfy users' demands calls for the development of computer programs able to produce text summaries. The process of automatically producing a summary from a source text consists of the following steps:

1. interpreting the text

2. extracting the relevant information, which ideally includes the "topics" of the source

3. condensing the extracted information and constructing a summary representation

4. presenting the summary representation to the reader in natural language.

Even though some approaches to text summarization produce acceptable summaries for specific tasks, it is generally agreed that the problem of coherent selection and expression of information in text summarization is far from being resolved. Sparck Jones and Endres-Niggemeyer (1995) stated the need for a research program in text

---

∗ Department of Computer Science, University of Sheffield, Sheffield, England, United Kingdom, S1 4DP. E-mail: saggion@dcs.shef.ac.uk

† Département d'Informatique et Recherche Opérationnelle, Université de Montréal, CP 6128, Succ Centre-Ville, Montréal, Québec, Canada, H3C 3J7. E-mail: lapalme@iro.umontreal.ca

1 In 1998, the volume of this information was calculated at somewhere between 400 and 500 million documents (Filman and Pant 1998).

summarization that would study the relation between source document and summary, the different types of summaries and their functions, the development of new methods and/or combination of already existing techniques for text summarization, and the development of evaluation procedures for summaries and systems. Rowley (1982) proposes the following typology of different types of document condensations:

- the **extract**, which is a set of passages selected from a source document to represent the whole document

- the **summary**, which occurs at the end of the document and is a restatement of the salient findings of a work

- the **abridgment**, which is a reduction of the original document that necessarily omits secondary points

- the **precis**, which stands for the main points of an argument

- the **digest**, which is a condensation of a book or news article

- the **highlight**, which is a comment included in specific parts of a document to alert a reader

- the **synopsis**, which in cinematography represents a script of a film.

In our research, we are concerned only with summaries of technical articles, which are called **abstracts**. In this context, two main types of abstracts are considered (ANSI 1979; ERIC 1980; Maizell, Smith, and Singer 1971): indicative abstracts, which point to information alerting the reader about the content of an article in a given domain (these abstracts will contain sentences like "The work of Consumer Advice Centres is examined."), and informative abstracts, which provide as much quantitative or qualitative information contained in the source document as possible (these abstracts will contain sentences like "Consumer Advice Centres have dealt with preshopping advice, education on consumers' rights and complaints about goods and services, advising the client and often obtaining expert assessments."). In the course of our research, we have studied the relation between abstracts and source documents, and as a result, we have developed SumUM (Summarization at Université de Montréal), a text summarization system that produces an indicative-informative abstract for technical documents. The abstracts are produced in two steps: First, the reader is presented with an indicative abstract that identifies the topics of the document (what the authors present, discuss, etc.). Then, if the reader is interested in some of the topics, specific information about them from the source document is presented in an informative abstract.

Figure 1 shows an automatic abstract produced by our system. The abstract was produced by a process of conceptual identification and text re-generation we call **selective analysis**. The indicative abstract contains information about the topic of the document. It describes the topics of sections and introduces relevant entities. The identified topics are terms either appearing in the indicative abstract or obtained from the terms and words of the indicative abstract through a process of term expansion. The one particular feature of these terms is that they can be used to obtain more conceptual information from the source document, such as definitions or statements of relevance, usefulness, and development, as can be seen in Figure 2.

This article is organized as follows. In the next section, we describe the analysis of a corpus of professional abstracts used to specify selective analysis; conceptual and linguistic information for the task of summarization of technical texts deduced from this corpus is also presented. An overview of selective analysis and the implementation

Designing for human-robot symbiosis

Presents the views on the development of intelligent interactive service robots. The authors have observed that a key research issue in service robotics is the integration of humans into the system. Discusses some of the technologies with particular emphasis on human-robot interaction, and system integration; describes human direct local autonomy (HuDL) in greater detail; and also discusses system integration and intelligent machine architecture (IMA). Gives an example implementation; discusses some issues in software development; and also presents the solution for integration, the IMA. Shows the mobile robot.

Identified Topics: **HuDL - IMA - aid systems - architecture - holonic manufacturing system - human - human-robot interaction - intelligent interactive service robots - intelligent machine architecture - intelligent machine software - interaction - key issue - widely used interaction - novel software architecture - overall interaction - robot - second issue - service - service robots - software - system - Technologies**

Figure 1
Indicative abstract and identified topics for the text "Designing for Human-Robot Symbiosis," D. M. Wilkes et al., *Industrial Robot*, 26(1), 1999, 49–58.

Development of a service robot is an extremely challenging task.
In the IRL, we **are using HuDL** to guide the development of a cooperative service robot team.
IMA is a two-level software architecture for rapidly integrating these elements, for an intelligent machine such as a service robot.
A holonic manufacturing system is a manufacturing system having autonomous but cooperative elements called holons (Koestler, 1971).
Communication between the robot and the human is **a key concern for intelligent service robotics**.

Figure 2
Informative abstract elaborating some topics.

of our experimental prototype, SumUM, is then presented in section 3. In section 4, we discuss the limitations of our approach; then, in section 5, we present an evaluation and comparison of our method with state-of-the art summarization systems and human abstracts. Related work on text summarization is discussed in section 6. Finally, in section 7, we draw our conclusions and discuss prospects for future research.

2. Observations from a Corpus

We have developed our method of text summarization by studying a corpus of professional abstracts and source documents. Our corpus contains 100 items, each composed of a professional abstract and its source document. As sources for the abstracts we used the journals *Library & Information Science Abstracts* (LISA), *Information Science Abstracts*

(ISA), and *Computer & Control Abstracts*. The source documents were found in journals of computer science (CS) and information science (IS), such as *AI Communications*, *AI Magazine*, *American Libraries*, *Annals of Library Science & Documentation*, *Artificial Intelligence*, *Computers in Libraries*, and *IEEE Expert*, among others (a total of 44 publications were examined). The professional abstracts contained three sentences on the average, with a maximum of seven and a minimum of one. The source documents covered a variety of subjects from IS and CS. We examined 62 documents in CS and 38 in IS, some of them containing author-provided abstracts. Most of the documents are structured in sections; but apart from conceptual sections such as "Introduction" and "Conclusion," they do not follow any particular style (articles from medicine, for example, usually have a fixed structure like "Introduction," "Method," "Statistical Analysis," "Result," "Discussion," "Previous Work," "Limitations," "Conclusion," but this was not the case in our corpus). The documents were 7 pages on average, with a minimum of 2 and a maximum of 45. Neither the abstracts nor the source documents were electronically available, so the information was collected through photocopies. Thus we do not have information regarding number of sentences and words in the source document.

Our methodological approach consisted of the manual alignment of sentences from the professional abstract with elements of the source document. This was accomplished by looking for a match between the information in the professional abstract and the information in the source document. The structural parts of the source document we examined were the title of the source document, the author abstract, the first section, the last section, the section headings, and the captions of tables and figures. When the information was not found, we looked in other parts of the source document. The information is not always found anywhere in the source document, in which case we acknowledge that fact. This methodological process was established after studying procedures for abstract writing (Cremmins 1982; Rowley 1982) and some initial observations from our corpus. One alignment is shown in Table 1. All alignments are available for research purposes at the SumUM Web page ⟨http://www-rali.iro.umontreal.ca/sumum.html⟩.

In this example, the three sentences of the professional abstract were aligned with four elements of the source document, two in the introduction and two in the author-provided abstract. The information of the abstract was found "literally" in the source document. The differences between the sentences of the professional abstract and those of the source document are the persons of the verbs ("Presents" vs. "We present" in alignment (1)), the verbs ("were discovered" vs. "We found" in alignment (3)), the impersonal versus personal styles ("Uses" vs. "Our experiment used" in alignment (2)), and the use of markers in the source document ("In this paper" in alignment (1)). This example shows that the organization of the abstract does not always mirror the organization of the source document.

2.1 Distributional Results

The 309 sentences of the professional abstracts in our corpus were manually aligned with 568 elements in the source documents. (We were not able to align six sentences of the professional abstracts.) Other studies have already investigated the alignment between sentences in the abstract and sentences in the source document. Kupiec, Pedersen, and Chen (1995) report on the semiautomatic alignment of 79% of sentences of professional abstracts in a corpus of 188 documents with professional abstracts. Using automatic means, it is difficult to deal with conceptual alignments that appeared in our corpus. Teufel and Moens (1998) report on a similar work, but this time on the alignment of sentences from author-provided abstracts. They use a corpus of 201 articles, obtaining only 31% of alignable sentences by automatic means. No informa-

Table 1
Item of corpus. Professional abstract: Library & Information Science abstract 3024 and source document: "Movement Characteristics Using a Mouse with Tactile and Force Feedback," *International Journal of Human-Computer Studies*, 45(5), October 1996, pages 483–493.

Ex.	Professional Abstract	Source Document	Position/Type
(1)	Presents the results of an empirical study that investigates the movement characteristics of a multi-modal mouse—a mouse that includes tactile and relevance feedback.	In this paper, we present the results of an empirical study that investigates the movement characteristics of a multi-modal mouse—a mouse that includes tactile and force feedback.	1st/Intr.
(2)	Uses a simple target selection task while varying the target distance, target size, and the sensory modality.	Our experiment used a simple target selection task while varying the target distance, target size, and the sensory modality.	1st/Intr.
(3)	Significant reduction in the overall movement times and in the time taken to stop the cursor after entering the target were discovered, indicating that modifying a mouse to include tactile feedback, and to a lesser extent, force feedback, offers performance advantages in target selecting tasks.	We found significant reductions in the overall movement time and in the time to stop the cursor after entering the target.	—/Abs.
		The results indicate that modifying a mouse to include tactile feedback, and to a lesser extent, force feedback, offers performance advantages in target selection tasks.	—/Abs.

Table 2
Distribution of information.

	Documents		With Author Abstract		Without Author Abstract		Average
	#	%	#	%	#	%	%
Title	10	2	6	2	4	1	2
Author abstract	83	15	83	34			20
First section	195	34	61	26	134	42	40
Last section	18	3	6	2	12	4	4
Headlines and captions	191	33	76	31	115	36	23
Other sections	71	13	13	5	58	17	11
Total	568	100	245	100	323	100	100

tion is given about the distribution of the sentences in structural parts in the source document.

In Table 2, we present the distribution of the sentences in the source documents that were aligned with the professional abstracts in our corpus. We consider all the structured documents of our corpus (97 documents). The first three columns contain the information for all documents, for documents with author abstracts, and for documents without author abstracts (the information is given in terms of total elements and

percentage of elements). We also recorded how the types of information are distributed in the professional abstract. For each abstract, we computed the ratio of the number of elements of each type contributing to the abstract to the total number of elements in the abstract (for example, the abstract in Table 1 contains 50% of first section and 50% of author abstract). The last column gives the average of the information over all abstracts. In this corpus, we found that 72% of the information for the abstracts comes from the following structural parts of the source documents: the title of the document, the first section, the last section, and the section headers and captions of tables and figures (sum of these entries on the first column of Table 2). Sharp (1989) reports on experiments carried out with abstractors in which it is shown that introductions and conclusions provide a basis for producing a coherent and informative abstract. In fact abstractors use a short cut strategy (looking at the introduction and conclusion) prior to looking at the whole paper. But our results indicate that using just those parts is not enough to produce a good informative abstract. Important information is also found in sections other than the introduction and conclusion. Abstractors not only select the information for the abstract because of its particular position in the source document, but they also look for specific types of information that happen to be lexically marked. In Table 1 the information reported is the topic of the document, the method, and the author's discovery. This information is lexically marked in the source document by expressions such as *we*, *paper*, *present*, *study*, *experiment*, *use*, *find*, and *indicate*. Based on these observations we have defined a conceptual and linguistic model for the task of text summarization of technical articles.

2.2 Conceptual Information for Text Summarization

A scientific and technical article is the result of the complex process of scientific inquiry, which starts with the identification of a problem and ends with its solution. It is a complex linguistic record of knowledge referring to a variety of real and hypothetical concepts and relations. Some of them are domain dependent (like diseases and treatments in medical science; atoms and fusion in physics; and algorithms and proofs in computer science), whereas others are generic to the technical literature (authors, the research article, the problem, the solution, etc.). We have identified 55 concepts and 39 relations that are typical of a technical article and relevant for identifying types of information for text summarization by collecting domain-independent lexical items and linguistic constructions from the corpus and classifying them using thesauri (Vianna 1980; Fellbaum 1998). We expanded the initial set with other linguistic constructions not observed in the corpus.

Concepts.   Concepts can be classified in categories referring to the authors (the authors of the article, their affiliation, researchers, etc.), the work of the authors (work, study, etc.), the research activity (current situation, need for research, problem, solution, method, etc.), the research article (the paper, the paper components, etc.), the objectives (objective, focus, etc.), and the cognitive activities (presentation, introduction, argument, etc.).

Relations.   Relations refer to general activities of the author during the research and writing of the work: studying (investigate, study, etc.), reporting the work (present, report, etc.), motivating (objective, focus, etc.), thinking (interest, opinion, etc.), and identifying (define, describe, etc.).

Types of Information.   We have identified 52 types of information for the process of automatic text summarization referring to the following aspects of the technical

Table 3
Conceptual information for text summarization.

Domain concepts	author, institutions, affiliation, author related, research group, project, research paper, others' paper, study, research, problem, solution, method, result, experiment, need, goal, focus, conclusion, recommendation, summary, researcher, work, hypothesis, research question, future plan, reference, acronym, expansion, structural, title, caption, quantity, mathematical, paper component, date, conceptual goal, conceptual focus, topic, introduction, overview, survey, development, analysis, comparison, discussion, presentation, definition, explanation, suggestion, discovery, situation, advantage, example
Domain relations	make known, show graphical material, study, investigate, summarize, situation, need, experiment, discover, infer, problem, solution, objective, focus, conclude, recommend, create, open, close, interest, explain, opinion, argue, comment, suggest, evidence, relevance, define, describe, elaborate, essential, advantage, use, identify entity, exemplify, effective, positive, novel, practical
Indicative types	topic of document, possible topic, topic of section, conceptual goal, conceptual focus, author development, development, inference, author interest, interest, author study, study, opening, closing, problem, solution, topic, entity introduction, acronym identification, signaling structure, signaling concept, experiments, methodology, explaining, commenting, giving evidence, need for research, situation, opinion, discovery, demonstration, investigation, suggestion, conclusion, summarization
Informative types	relevance, goal, focus, essential, positiveness, usefulness, effectiveness, description, definition, advantage, practicality, novelty, elaboration, exemplification, introduction, identification, development

article: background information (situation, need, problem, etc.), reporting of information (presenting entities, topic, subtopics, objectives, etc.), referring to the work of the author (study, investigate, method, hypothesis, etc.), cognitive activities (argue, infer, conclude, etc.), and elaboration of the contents (definitions, advantages, etc.).

The complete list of concepts, relations, and types of information is provided in Table 3. Concepts and relations are the basis for the classification of types of information referring to the essential contents of a technical abstract. Nevertheless, the presence of a single concept or relation in a sentence is not enough to understand the type of information it conveys. The co-occurrence of concepts and relations in appropriate linguistic-conceptual patterns is used in our case as the basis for the classification of the sentences. The types of information are classified as **Indicative** or **Informative** depending on the type of abstract to which they will contribute. For example, **Topic of Document** and **Topic of Section** are indicative, whereas **Goal of Entity** and **Description of Entity** are informative. Note that we have identified only a few linguistic expressions used to express particular elements of the conceptual model, because we were mainly concerned with the development of a general method of text summarization and because the task of constructing such linguistic resources is time consuming.

2.3 From Source to Abstract
According to Cremmins (1982), the last step in the human production of the summary text is the "extracting" into "abstracting" step in which the extracted information will be mentally sorted into a preestablished format and will be "edited" using cognitive techniques. The editing of the raw material ranges from minor to major operations. Cremmins gives little indication, however, about the process of editing.

Table 4
Text editing in human abstracting.

Professional Abstract	Source Document
Mortality in rats and mice of both sexes was dose related.	There were significant positive associations between the concentrations of the substance administered and mortality in rats and mice of both sexes.
No treatment related tumors were found in any of the animals.	There was no convincing evidence to indicate that endrin ingestion induced any of the different types of tumors which were found in the treated animals.

Major transformations are those of the complex process of language understanding and production, such as deduction, generalization, and paraphrase. Some examples of editing given by Cremmins are shown in Table 4. In the first example, the concept *mortality in rats and mice of both sexes* is stated with the wording of the source document; however, the concept expressed by *the concentrations of the substance administered* is stated with the expression *dose*. In the second example, the relation between the *tumors* and *endrin ingestion* is expressed through the complex nominal *treatment related tumors*.

In his rules for abstracting, Bernier (1985) states that redundancy, repetition, and circumlocutions are to be avoided. He gives a list of linguistic expressions that can be safely removed from extracted sentences or reexpressed in order to gain conciseness. These include expressions such as *It was concluded that X*, to be replaced by *X*, and *It appears that*, to be replaced by *Apparently*. Also, Mathis and Rush (1985) indicate that some transformations in the source material are allowed, such as concatenation, truncation, phrase deletion, voice transformation, paraphrase, division, and word deletion. Rowley (1982) mentions the inclusion of the lead or topical sentence and the use of active voice and advocates conciseness. But in fact, the issue of editing in text summarization has usually been neglected, notable exceptions being the works by Jing and McKeown (2000) and Mani, Gates, and Bloedorn (1999). In our work, we partially address this issue by enumerating some transformations frequently found in our corpus that are computationally implementable. The transformations are always conceptual in nature and not textual (they do not operate on the string level), even if some of them seem to take the form of simple string deletion or substitution. The rephrasing transformations we have identified are outlined below. We also include for each transformation the number and percentage of times the transformation was used to produce a sentence of the professional abstract. (Note that the percentages do not add up to 100, as sentences can be involved in more than one operation.)

> *Syntactic verb transformation:* Some verbs from the source document are reexpressed in the abstract, usually in order to make the style impersonal. The person, tense, and voice of the original verb are changed. Also, verbs that are used to state the topic of the document are generally expressed in the present tense (in active or passive voice). The same applies to verbs introducing the objective of the research paper or investigation (according to convention, objectives are reported in the present tense and results in the past tense). This transformation was observed 48 times (15%).

Lexical verb transformation: A verb used to introduce a topic is changed and restated in the impersonal form. This transformation was observed 13 times (4%).

Verb selection: The topic or subtopic of the document is introduced by a domain verb, usually when information from titles is used to create a sentence. This transformation was observed 70 times (21%).

Conceptual deletion: Domain concepts such as **research paper** and **author** are avoided in the abstract. This transformation was observed 43 times (13%).

Concept reexpression: Domain concepts such as **author**, **research paper**, and **author-related entity** are stated in the impersonal form. This transformation was observed 4 times (1%).

Structural deletion: Discourse markers (contrast, structuring, logical consequence, adding, etc.) such as *first*, *next*, *finally*, *however*, and *although* are deleted. This transformation was observed 7 times (2%).

Clause deletion: One or more clauses (principal or complement) of the sentence are deleted. This transformation was observed 47 times (14%).

Parenthetical deletion: Some parenthetical expressions are eliminated. This transformation was observed 10 times (3%).

Acronym expansion: Acronyms introduced for the first time are presented along with their expansions, or only the expansion is presented. This transformation was observed 7 times (2%).

Abbreviation: A shorter expression (e.g., acronym or anaphoric expression) is used to refer to an entity. This transformation was observed 3 times (1%).

Merge: Information from several parts of the source document are merged into a single sentence. This is the usual case when reporting entities stated in titles and captions. This transformation was observed 124 times (38%).

Split: Information from one sentence of the source document is presented in separate sentences in the abstract. This transformation was observed 3 times (1%).

Complex reformulation: A complex reformulation takes place. This could involve several cognitive processes, such as generalization and paraphrase. This transformation was observed 75 times (23%).

Noun transformations: Other transformations take place, such as nominalization, generalization, restatement of complex nominals, deletion of complex nominals, expansion of complex nominals (different classes of aggregation), and change of initial uppercase to lowercase (e.g., when words from titles or headlines, usually in upper initial, are used for the summary). This transformation was observed 70 times (21%).

No transformation: The information is reported as in the source. This transformation was observed 35 times (11%).

We found that transformations involving domain verbs appeared in 40% of the sentences, noun editing occurred in 38% of the sentences, discourse level editing occurred in 19% of the sentences, merging and splitting of information occurred in 38% of the sentences, complex reformulation accounts for 23% of the sentences, and finally,

only 11% of the information from the source document is stated without transformation. Although most approaches to automatic text summarization present the extracted information in both the order and the form of the original, this is not the case in human-produced abstracts. Nevertheless, some transformations in the source document could be implemented by computers with state-of-the-art techniques in natural language processing in order to improve the quality of the automatic summaries.

In this section, we have studied relations between abstracts and their source documents. This study was motivated by the need to answer to the question of content selection in text summarization (Sparck Jones 1993). We have also addressed here another important research question: how the information is expressed in the summary. Our study was based on the manual construction of alignments between sentences of professional abstracts and elements of source documents. In order to obtain an appropriate coverage, abstracts from different secondary sources and source documents from different journals were used. We have shown that more than 70% of the information for abstracts comes from the introduction, conclusion, titles, and captioning of the source document. This is an empirical verification of what is generally acknowledged in practical abstract writing in professional settings. We have also identified 15 types of transformation usually applied to the source document in order to produce a coherent piece of text. Of the sentences of our corpus, 89% have been edited. In section 3.1, we detail the specification of patterns of sentence and text production inspired from our corpus study that were implemented in our automatic system.

Although the linguistic information for our model has been manually collected, Teufel (1998) has shown how this labor-intensive task can be accomplished in a semi-automatic fashion. The analysis presented here and the idea of the alignments have been greatly influenced by the exploration of abstracting manuals (Cremmins 1982). Our conceptual model comes mainly from the empirical analysis of the corpus but has also been influenced by work on discourse modeling (Liddy 1991) and in the philosophy of science (Bunge 1967). It is interesting to note that our concerns regarding the presentation and editing of the information for text summarization are now being addressed by other researchers as well. Jing and McKeown (2000) and Jing (2000) propose a **cut-and-paste** strategy as a computational process of automatic abstracting and a sentence reduction strategy to produce concise sentences. They have identified six "editing" operations in human abstracting that are a subset of the transformation found in our study. Jing and McKeown's work on sentence reduction will be discussed in section 6. Knight and Marcu (2000) propose a noisy-channel model and a decision-based model for sentence reduction also aiming at conciseness.

3. Selective Analysis and Its Implementation

Selective analysis is a method for text summarization of technical articles whose design is based on the study of the corpus described in section 2. The method emphasizes the selection of particular types of information and its elaboration, exploring the issue of dynamic summarization. It is independent of any particular implementation. Nevertheless, its design was motivated by actual needs for accessing the content of long documents and the current limitations of natural language processing of domain-independent texts. Selective analysis is composed of four main steps, which are briefly motivated here and fully explained in the rest of the section.

- *Indicative selection:* The function of indicative selection is to identify potential topics of the document and to instantiate a set of indicative templates. These templates are instantiated with sentences matching

specific patterns. A subset of templates is retained based on a matching process between terms from titles and terms from the indicative templates. From the selected templates, terms are extracted for further analysis (i.e., potential topics).

- *Informative selection:* The information selection process determines the subset of topics computed by the indicative selection that can be informatively expanded according to the interest of the reader. This process considers sentences in which informative markers and interesting topics co-occur and instantiates a set of informative templates that elaborate the topics.

- *Indicative generation:* In indicative generation, the set of templates detected by the indicative selection are first sorted using a preestablished conceptual order. Then, the templates are used to generate sentences according to the style observed in the corpus of professional abstracts (i.e., verbs in the impersonal and reformulation of some domain concepts). When possible, information from different templates is integrated in order to produce a single sentence. A list of topics is also presented to the reader.

- *Informative generation:* In informative generation, the reader selects some of the topics presented as a result of indicative generation, thereby asking for more information about those topics. Templates instantiated by the informative selection associated with the selected topics are used to present additional information to the reader.

Whereas the indicative abstract depends on the structure, content, and to some extent, on specific types of information generally reported in this kind of summary, the informative abstract relies on the interests of the reader to determine the topics to expand.

3.1 Implementing SumUM

The architecture of SumUM is depicted in Figure 3. Our approach to text summarization is based on a superficial analysis of the source document to extract appropriate types of information and on the implementation of some text regeneration techniques. SumUM has been implemented in SICStus Prolog (release 3.7.1) (SICStus 1998) and Perl (Wall, Christiansen, and Schwartz 1996) running on Sun workstations (5.6) and Linux machines (RH 6.0). For a complete description of the system and its implementation, the reader is referred to Saggion (2000).

The sources of information we use for implementing our system are a POS tagger (Foster 1991); linguistic and conceptual patterns specified by regular expressions combining POS tags, our syntactic categories, domain concepts, and words; and a conceptual dictionary that implements our conceptual model (241 domain verbs, 163 domain nouns, and 129 adjectives); see Table 5.

3.1.1 Preprocessing and Interpretation. The input article (plain ASCII text in English without markup) is segmented in main units (title, author information, main sections and references) using typographic information (i.e., nonblank lines ending with a character different from punctuation surrounded by blank lines) and some keywords like "Introduction" and "References." Each unit is passed through the statistical tagger (based on bigrams). A scanning process reads each element of the tagged files and
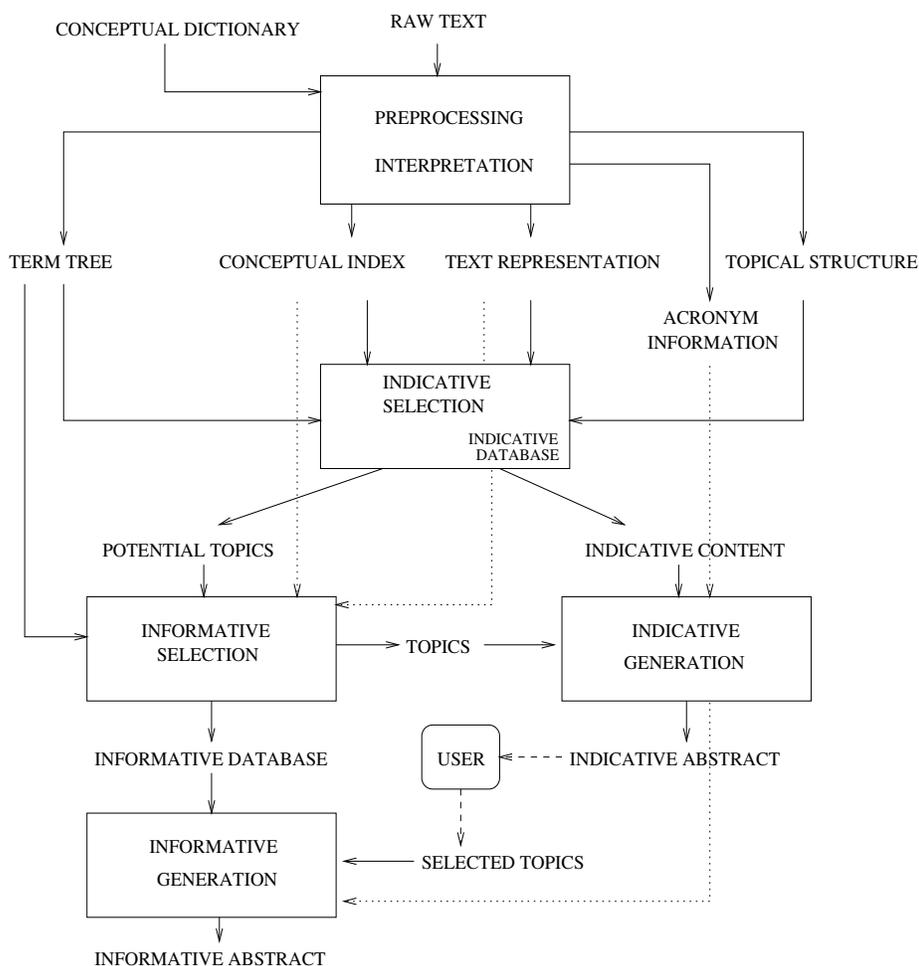
CONCEPTUAL DICTIONARY   RAW TEXT

PREPROCESSING

INTERPRETATION

TERM TREE   CONCEPTUAL INDEX   TEXT REPRESENTATION   TOPICAL STRUCTURE

ACRONYM
INFORMATION

INDICATIVE
SELECTION

INDICATIVE
DATABASE

POTENTIAL TOPICS   INDICATIVE CONTENT

INFORMATIVE
SELECTION   TOPICS   INDICATIVE
GENERATION

INFORMATIVE DATABASE   USER   INDICATIVE ABSTRACT

INFORMATIVE
GENERATION   SELECTED TOPICS

INFORMATIVE ABSTRACT

Figure 3
SumUM architecture.

transforms sequences of tagged words into lists of elements, each element being a list of attribute-value pairs. For instance, the word *systems*, which is a common noun is represented with the following attributes **(cat,'NomC')**, **(Nbr,plur)**, **(canon,system)** in addition to the original word. The frequency of each noun (proper or common) is also computed. SumUM gradually determines the paragraph structure of the document, relying on end of paragraph markers. Sentences are interpreted using finite-state transducers we developed (implementing 334 linguistic and domain-specific patterns) and the conceptual dictionary. The interpretation process produces a partial representation that consists of the sentence position (section and sentence numbers) and a list of syntactic constituents annotated with conceptual information. As title and section headers are recognized by position (i.e., sentence number 0 of the section), only noun group identification is carried out in those components. Each sentence constituent is represented by a list of attribute-value pairs. The parse of each element is as follows:

- *Noun group parsing.* We identify only nonrecursive, base noun groups. The parse of a noun group contains information about the original string, the canonical or citation form, syntactic features, the semantics

Table 5
Overview of the conceptual dictionary.

Concept/Relation	Lexical Item
make known	cover, describe, examine, explore, present, report, overview, outline, . . .
create	create, construct, ideate, develop, design, implement, produce, project, . . .
study	investigate, compare, analyze, measure, study, estimate, contrast, . . .
interest	address, interest, concern, matter, worry, . . .
infer	demonstrate, infer, deduce, show, conclude, draw, indicate, . . .
identify entity	include, classify, call, contain, categorize, divide, . . .
paper	paper, article, report, . . .
paper component	section, subsection, appendix, . . .
structural	figure, table, picture, graphic, . . .
problem	complexity, intricacy, problem, difficulty, lack, . . .
goal	goal, objective, . . .
result	finding, result, . . .
important	important, relevant, outstanding, . . .
necessary	needed, necessary, indispensable, mandatory, vital, . . .
novelty	innovative, new, novel, original, . . .

  (i.e., the head of the group in citation form), adjectives, and information
  referring to the conceptual model that is optional.

- *Verb group parsing.* The parse of a verb group contains information about
  the original string, the semantics (i.e., the head of the group in citation
  form), the syntactic features, information about adverbs, and the
  conceptual information that is optional.

- *Adjectives and adverbials.* The parse of adjectival and adverbial groups
  contains the original string, the citation form, and the optional
  information from the conceptual model.

- *Other.* The rest of the elements (i.e., conjunctions, prepositions, etc.) are
  left unanalyzed.

In order to assess the accuracy of the parsing process, we manually extracted base
noun groups and base verb groups from a set of 42 abstracts found on the INSPEC
(2000) service (about 5,000 words). Then, we parsed the abstracts and automatically
extracted noun groups and verb groups with our finite-state machinery and computed
recall and precision measures. Recall measures the ratio of the number of correct
syntactic constructions identified by the algorithm to the number of correct syntactic
constructions. Precision is the ratio of the number of correct syntactic constructions
identified by the algorithm to the total number of constructions identified by the
algorithm. We found the parser to perform at 86% recall and 86% precision for noun
groups and 85% recall and 76% precision for verb groups.

Term extraction.   Terms are constructed from the citation form of noun groups.
They are extracted from sentences and stored along with their semantics and position
in the **term tree**, an AVL tree structure for efficient access from the SICStus Prolog
association lists package. As each term is extracted from a sentence, its frequency is
updated. We also build a **conceptual index** that specifies the types of information of
each sentence using the concepts and relations identified before. Finally, terms and

Table 6
Specification of indicative templates for the topic of the document and the topic of a section.

Type:	topic
Id:	integer identifier
Predicate:	instance of **make known**
Where:	instance of {**research paper, study, work, research**}
Who:	instance of {**research paper, author, study, work, research**}
What:	parsed sentence fragment
Position:	section and sentence id
Topic candidates:	list of terms from the **What** filler
Weight:	number

Type:	sec_desc
Id:	integer identifier
Predicate:	instance of **make known**
Section:	instance of **section(Id)**
Argument:	parsed sentence fragment
Position:	section and sentence id
Topic candidates:	list of terms from the **Argument** filler
Weight:	number

words are extracted from titles (identified as those sentences with numeral 0 in the representation) and stored in a list, the **topical structure**, and acronyms and their expansions are identified and recorded.

3.1.2 Indicative Selection. Simple templates are used to represent the types of information. We have implemented 21 indicative templates in this version of SumUM. Table 6 presents two of these indicative templates and their slots. The slot **Topic candidates** is filled with terms and acronym expansions. Term relevance is the total frequency of all nominal components of the term divided by the total number of nominal components. It is computed using the following formula:

$$\text{relevance(Term)} = \frac{\sum_{\{N \in \text{Term} \wedge \text{ noun(N)}\}} \text{noun\_frequency}(N)}{|N : N \in \text{ Term} \wedge \text{ noun}(N)|}$$

where $noun(N)$ is true if N is a noun, $noun\_frequency(N)$ is a function computed during preprocessing and interpretation that gives the word count for noun N, and the notation $|S|$ stands for the number of elements in the set S. As complex terms have lower distribution than single terms, this formula gives us an estimate of the distribution of the term and its components in the document. In doing so, a low-frequency term like *robot architecture* is assigned a high degree of relevance because chances are that *robot* and *architecture* occur frequently on their own. Other techniques exist for boosting the score of longer phrases, such as adjusting the score of the phrase by a fixed factor that depends on the length of the phrase (Turney 1999). The **Weight** slot is filled in with the sum of the relevance of the terms on the **Topic candidates** slot.

For determining the content of the indicative abstract, SumUM considers only sentences that have been identified as carrying indicative information; excludes sentences containing problematic anaphoric references ("the first...," "the previous...," "that...," quantifiers in sentence initial, etc.), those that are not domain concepts (e.g., "These results," "The first section," etc.), and some connectives ("although," "however," etc.); and checks whether the sentence matches an indicative pattern. Indicative

Table 7
Indicative pattern specification and sentence fragments matching the patterns (in parentheses).

Signaling structural	$SKIP_1$ + **GN** + **Prep** + **GN** + **show graphical material** + **Prep** + **structural** (*In our case, the architecture of the self-tuner is shown in Figure 3 Auto-tuning...*)
Topic	$SKIP_1$ + **research paper** + $SKIP_2$ + **author** + *ARGUMENT* (*In this article, we overview the main techniques used in order...*)
Author's Goal	*SKIP* + **conceptual goal** + *SKIP* + **define** + *GOAL* (*Our goals within the HMS project are to develop a holonic architecture for...*)
Signaling concept	*SKIP* + **development** + **Prep** + **GN** (*Implementation of industrial robots*)
Section Topic	**paper component** + **make known** + *ARGUMENT* + **ConC** + **paper component** (*Section 2 describes HuDL in greater detail and Section 3...*)
Problem/Solution	*SKIP* + **solution (dr)** + **problem** (*The proposed methodology overcomes the problems caused by...*)
Introduce Entity	**GN** + **define** + *SKIP* (*Rapid Prototyping (RP) is a technique...*)

patterns contain variables, syntactic constructions, domain concepts, and relations. One hundred seventy-four indicative patterns have been implemented; some of them are shown in Table 7.

For each matched pattern, SumUM verifies some restrictions, such as verb tenses and voice, extracts information from pattern variables, and instantiates a template of the appropriate type. All the instantiated templates constitute the indicative database (IDB). SumUM matches each element of the topical structure with the terms of the **Topic candidate** slots of templates in the IDB. Two terms $Term_1$ and $Term_2$ match if $Term_1$ is a substring of $Term_2$ or if $Term_2$ is a substring of $Term_1$ (e.g., *robotic fruit harvester* matches *harvester*).

Then, SumUM selects the template with the greatest **Weight**. In case of conflict, types are selected following the precedence given in Table 8. This order gives preference to explicit topical information more usually found in indicative abstracts. Where there is conflict, the **Position** and the **Id** slots are used to decide: If two topic templates have the same **Weight**, the template with position closer to the beginning of the document is selected, and if they are still equal, the template with lower **Id** is used. SumUM prioritizes topical information by selecting the topical template with greatest weight. The selected templates constitute the **indicative content** (IC), and the terms and words appearing in the **Topic candidate** slots and their expansions constitute the **potential topics** (PTs) of the document. Expansions are obtained by looking for terms in the term tree sharing the semantics of any term in the IC.

3.1.3 Informative Selection. For each potential topic PT and sentence in which it appears, SumUM checks whether the sentence contains an informative marker and matches a dynamic informative pattern. Dynamic patterns include a *TOPIC* slot instantiated with the PT before trying a match. They also include concepts, relations, and linguistic information. Eighty-seven informative patterns have been implemented,

Table 8
Precedence for content selection.

Topic of Document > **Topic of Section** > **Topic Description** > **Possible Topic** > **Author Study** > **Author Development** > **Author Interest** > **Conceptual Goal, Research Goal** > **Conceptual Focus, Focus** > **Entity Introduction** > **Entity Identification** > **Signaling Structural, Signaling Concepts** > **Other Indicative Types**

Table 9
Informative pattern specification and sentence fragments matching the patterns (in parentheses).

Definition	*SKIP* + *TOPIC* + **define** + **GN** (*The RIMHO walking robot is a prototype developed with the aim of...*)
Description	*SKIP* + *TOPIC* + **describe** (*The hardware of the MMI consists of a main pendant (MP), an operator pendant...*)
Use	*SKIP* + **use** + *TOPIC* (*To realize the control using an industrial robot, such as...*)
Advantage	*SKIP* + **advantage** + **Prep** + *TOPIC* (*The biggest advantage of SWERS is the easier and faster...*)
Effectiveness	*SKIP* + *TOPIC* + **define** + **effective** (*The system is effective in the task of...*)

some of which are presented in Table 9. If a sentence satisfies an informative pattern, the PT is considered a topic of the document, and an informative template is instantiated with the sentence. The informative templates contain a **Content** slot to record the information from the sentence, a **Topic** slot to record the topic, and a **Position** slot to record positional information. Examples are presented in Tables 10 and 11. The templates obtained by this process constitute the Informative Data Base (InfoDB), and the topics are the terms appearing in the slot **Topic** of the templates in the InfoDB.[2]

3.1.4 Generation. The process of generation consists of the arrangement of the information in a preestablished conceptual order, the merging of some types of information, and the reformulation of the information in one text paragraph. The IC is sorted using positional information and the order presented in Table 12, which is typical of technical articles.

SumUM merges groups of up to three templates of type **Topic of Document** to produce more complex sentences (**Merge** transformation). The same is done for templates of type **Topic of Section**, **Signaling Concept**, and **Signaling Structural**. The template **Signaling Concept** contains information about concepts found on section headings; SumUM selects an appropriate verb to introduce that information in the abstract (**Verb Selection**). In this way, for example, given the section heading "Experimental Results," SumUM is able to produce the sentence "Presents experimental results."

The sorted templates constitute the **text plan**. Each element in the text plan is used to produce a sentence the structure of which depends on the template. The schema of presentation of a text plan composed of $n(\geq 1)$ templates $Tmpl_i$ is as follows:

$$Text = \bigoplus_{i=1}^{n}[\overline{Tmpl_i} \oplus \text{"."}].$$

The notation $\bar{A}$ means the string produced by the generation of A, $\oplus$ denotes concatenation, and $\bigoplus_{i=1}^{n} A_i$ stands for the concatenation of all A_i. We assume that all the parameters necessary for the generation are available (i.e., voice, tense, number, position, etc.).

The schema of presentation of a template $Tmpl$ of type **Topic of the Document** is:[3]

$$\overline{Tmpl} = \overline{Tmpl.Predicate} \oplus \overline{Tmpl.What}$$

---

2 *TOPIC* = {*Term* : ∃*Template* ∈ *InfoDB* ∧ *Template.Topic* = *Term*}.
3 The notation *Tmpl.Slot* denotes the content of slot *Slot* of template *Tmpl*.

Table 10
Specification of the templates for the description and definition of a topic.

Type:	description
Id:	integer identifier
Topic:	term
Predicate:	instance of **describe** (i.e., *X is composed* of *Y*)
Content:	parsed sentence fragment
Position:	section and sentence id

Type:	definition
Id:	integer identifier
Topic:	term
Predicate:	instance of **define** (i.e., *X is* a *Y*)
Content:	parsed sentence fragment
Position:	section and sentence id

Table 11
Definition template instantiated with sentence "REVERSA is a dual viewpoint noncontact laser scanner which comes complete with scanning software and data manipulation tools."

Type:	**definition**
Id:	41
Topic:	*REVERSA*
Predicate:	**be**, . . .
Content:	*REVERSA is a dual viewpoint noncontact laser scanner which. . .*
Position:	**Sentence 1 from Section 2**

The predicate is generated in the present tense of the third-person singular (**Syntactic Verb Transformation**). So sentences like "*X* will be presented" or "*X* have been presented" or "We have presented here *X*," which are usually found in source documents, will be avoided because they are awkward in an abstract. Arguments are generated by a procedure that expands/abbreviates acronyms (**Acronym Expansion** and **Abbreviation**), presents author-related entities in the impersonal form (**concept reexpression**), uses fixed expressions in order to refer to the authors and the research paper, and produces correct case and punctuation. Examples of sentences generated by the system have been presented in Saggion and Lapalme (2000a). In this way we implement some of the transformations studied in section 2.3. The schema of presentation of the

Table 12
Conceptual order for content expression.

Problem Solution, **Problem Identification**, **Need** and **Situation** in positional order
Topic of Document sorted in descending order of **Weight**
Possible Topic sorted in descending order of **Weight**
Topic Description, **Study**, **Interest**, **Development**, **Entity Introduction**, **Research Goal**,
   Conceptual Goal, **Conceptual Focus** and **Focus** in positional order
Method and **Experiment** in positional order
Results, **Inference**, **Knowledge** and **Summarization** in positional order
Entity Identification in positional order
Topic of Section in section order
Signaling Structural and **Signaling Concepts** in positional order

Topic of Section is

$$\overline{Tmpl} = \overline{Tmpl.Predicate} \oplus \overline{Tmpl.Argument}.$$

The schema of generation of a merged template *Tmpl* is

$$\overline{Tmpl} = \left(\bigoplus_{i=1}^{n-1} [\overline{Tmpl.Templates_i} \oplus \text{``;''}] \right) \oplus \text{``and also''} \oplus \overline{Tmpl.Templates_n},$$

where *Tmpl.Templates$_i$* is the *i*th template in the merge. Note that if *n* adjacent templates in the merge share the same predicate, then only one verb is generated, and the arguments are presented as a conjunction (i.e., "Presents *X* and *Y*." instead of "Presents *X* and presents *Y*."). This is specified with the following schema:

$$\overline{Tmpl} = \overline{Predicate} \oplus \overline{Tmpl_1.Arg} \oplus \left(\bigoplus_{i=2}^{n-1} [\overline{Tmpl_i.Arg} \oplus \text{``;''}] \right) \oplus \text{``and''} \oplus \overline{Tmpl_n.Arg},$$

where *Predicate* is the predicate common to the merged templates.

The indicative abstract is presented along with the list of topics that are obtained from the list **Topics**. SumUM presents in alphabetical order the first superficial occurrence of the term in the source document (this information is found in the term tree). For the informative abstract, the system retrieves from the InfoDB those templates matching the topics selected by the user (using the slot **Topic** for that purpose) and presents the information on the **Content** slots in the order of the original text (using the **Position** for that purpose).

4. Limitations of the Approach

Our approach is based on the empirical examination of abstracts published by second services and on assumptions about technical text organization (Paice 1991; Bhatia 1993; Jordan 1993, 1996). In our first study, we examined 100 abstracts and source documents in order to deduce a conceptual and linguistic model for the task of summarization of technical articles. Then we expanded the corpus with 100 more items in order to validate the model. We believe that the concepts, relations, and types of information identified account for interesting phenomena appearing in the corpus and constitute a sound basis for text summarization. The conceptual information has not been formalized in ontological form, opening an avenue for future developments. All the knowledge of the system (syntactic and conceptual) was manually acquired during specification, implementation, and testing. The coverage and completeness of the model have not been assessed in this work and will be the subject of future studies. Nevertheless SumUM has been tested in different technical domains.

The implementation of our method relies on noun and verb group identification, conceptual tagging, pattern matching, and template instantiation we have developed for the purpose of this research. The interpreter relies on the output produced by a shallow text segmenter and on a statistical part-of-speech tagger. Our prototype analyzes sentences for the specific purpose of text summarization and implements some patterns of generation observed in the corpus, including the reformulation of verb groups and noun groups, sentence combination or fusion, and conceptual deletion, among others. We have not addressed here the question of text understanding: SumUM is able to produce text summaries, but it is not able to demonstrate intelligent behavior

(answering questions, paraphrasing, anaphora resolution, etc.). Concerning the problem of text coherence, we have not properly addressed the problem of identification of anaphoric expressions in technical documents: SumUM excludes from the content of the indicative abstract sentences containing expressions considered problematic. The problem of anaphoric expressions in technical articles has been extensively addressed in research work carried out under the British Library Automatic Abstracting Project (BLAB) (Johnson et al. 1993; Paice et al. 1994). Although some of the exclusion rules implemented in the BLAB project are considered in SumUM (exclusion of sentences with quantifier subject, sentences with demonstratives, some initial connectives, and pronouns), our approach lacks coverage of some important cases dealt with in the BLAB rules, such as the inclusion of sentences because of dangling anaphora.

This implementation of SumUM ignores some aspects of the structure of textlike lists and enumerations, and most of the process overlooks the information about paragraph structure. Nevertheless, in future improvements of SumUM, these will be taken into consideration to produce better results.

5. Evaluating the Summaries

Abstracts are texts used in tasks such as assessing the content of a source document and deciding if it is worth reading. If text summarization systems are designed to fulfill the requirements of those tasks, the quality of the generated texts has to be evaluated according to their intended function. The quality of human-produced abstracts has been examined in the literature (Grant 1992; Kaplan et al. 1994; Gibson 1993), using linguistic criteria such as cohesion and coherence, thematic structure, sentence structure, and lexical density; in automatic text summarization, however, such detailed analysis is only just emerging. Content evaluation assesses whether an automatic system is able to identify the intended "topics" of the source document. Text quality evaluation assesses the readability, grammar, and coherence of a summary. The evaluations can be made in intrinsic or extrinsic fashions as defined by Sparck Jones and Galliers (1995).

An intrinsic evaluation measures the quality of the summary itself by comparing the summary with the source document, by measuring how many "main" ideas of the source document are covered by the abstract, or by comparing the content of the automatic summary with an ideal abstract (gold standard) produced by a human (Mariani 1995). An extrinsic evaluation measures how helpful a summary is in the completion of a given task. For example, given a document that contains the answers to some predefined questions, readers are asked to answer those questions using the document's abstract. If the reader correctly answers the questions, the abstract is considered of good quality for the given question-answering task. Variables measured can be the number of correct answers and the time to complete the task. Recent experiments (Jing et al. 1998) have shown how different parameters such as the length of the abstract can affect the outcome of the evaluation.

5.1 Evaluation of Indicative Content and Text Quality
Our objective in the evaluation of indicative content is to see whether the abstracts produced by our method convey the essential content of the source documents in order to help readers complete a categorization task. In the evaluation of text quality, we want to determine whether the abstracts produced by our method are acceptable according to a number of acceptability criteria.

5.1.1 Design. In both evaluations we are interested in comparing our summaries with summaries produced using other methodologies, including human-written ones. In order to evaluate the content, we presented evaluators with abstracts and five descriptors (lists of keywords) for each abstract. The evaluators had to find the correct descriptor for the abstract. One of the descriptors was the correct descriptor of the abstract and the others were descriptors from the same domain, obtained from the journals in which the source documents were published. In order to evaluate text quality, we asked the evaluators to provide an acceptability score between 0–5 for the abstract (0 for unacceptable and 5 for acceptable) based on the following criteria taken from Rowley (1982): good spelling and grammar, clear indication of the topic of the source document, conciseness, readability and understandability, and whether acronyms are presented along with their expansions. We told the evaluators that we would consider the abstracts with scores above 2.5 acceptable; with this information, they could use scores below or above that borderline to enforce acceptability. The design of this experiment was validated by three IS specialists. The experiment was run three times with different data each time and with a different set of summarizers (human or automatic). When we first designed this experiment, only one text summarization system was available to us, so we performed the experiment comparing automatic abstracts produced by two summarizers and abstracts published with the source documents. Later on, we found two other summarizers, and we decided to repeat the experiment only considering three automatic systems.

Our evaluation mirrors the TIPSTER SUMMAC categorization task (Firmin and Chrzanowski 1999; Mani et al. 1998) in which given a generic summary (or a full document), the human participant chooses a single category (out of five categories) to which the document is relevant. The evaluation seeks to determine whether the summary is effective in capturing whatever information in the document is needed to correctly categorize the document. In the TIPSTER SUMMAC evaluation, 10 Text Retrieval Conference (TREC) topics and 100 documents per topic were used, and 16 systems participated. The results for TREC indicate that there are no significant differences among the systems for the categorization task and that the performance using the full document is not much better.

5.1.2 Subjects and Materials. All our evaluators were IS students/staff from Université de Montréal, McGill University, and John Abbott College. They were chosen because they have knowledge about what constitutes a good indicative abstract. We used the Latin square experimental design, whereby forms included n abstracts from n different documents, where n depends on the number of subjects (thus an evaluator never compared different summaries of the same document). Each abstract was printed on a different page including the five descriptors, a field to be completed with the quality score associated with the abstract, and a field to be filled with comments about the abstract. In order to produce the evaluation forms, we used source documents (all technical articles) from the journal *Industrial Robots*, found in the Emerald Electronic Library ⟨http://www.emerald-library.com⟩. In addition to the abstracts published with source documents, we produced automatic abstracts using the following systems: SumUM, Microsoft'97 Autosummarize, Extractor, and n-STEIN. Microsoft'97 Autosummarize is distributed with Word'97. Extractor (Turney 1999) is a system that takes a text file as input (plain ASCII text, HTML, or e-mail) and generates a list of keywords and keyphrases as output. On average, it generates the number of phrases requested by the user, but the actual number for any given document may be slightly below or above the requested number, depending mainly on the length of the input document. Extractor has 12 parameters relevant for keyphrase extraction that are tuned by a genetic

algorithm to maximize performance on training data. We used Extractor 5.1, which is distributed for demonstration (downloaded from ⟨http://extractor.iit.nrc.ca/⟩). n-STEIN is a commercial system that was available for demonstration purposes at the time we were conducting our research (n-STEIN 2000) (January 2000). The system is based on a combination of statistical and linguistic processing. Unfortunately no technical details of the system are given.

5.1.3 Procedure. Each abstract was evaluated by three different evaluators, who were not aware of the method used to produce the abstracts. In order to measure the outcome of the categorization task, we considered the abstract to have helped in categorizing the source document if two or more evaluators were able to chose the correct descriptor for the abstract. In order to measure the quality of the abstract, we computed the average quality using the scores given by the evaluators.

5.1.4 Results and Discussion. In Table 13 we present the average information for three runs of this experiment. "Success" refers to the percentage of cases in which subjects identified the correct descriptor. "Quality" refers to subjects' summary quality score. Note that because of this particular design, we cannot compare numbers across experiments, we can only discuss results for each experiment.

Overall, for each experiment no significant differences were observed between the different automatic systems in the categorization task. All automatic methods performed similarly, though we believe that documents and descriptors of narrower domains are needed in order to correctly assess the effectiveness of each summarization method. Unfortunately, the construction of such resources goes beyond our present research and will be addressed in future work.

The figures for text acceptability indicate that abstracts produced by Autosummarize are below the acceptability level of 2.5. The abstracts produced by SumUM, Extractor, and n-STEIN are above the acceptability level of 2.5, and the human abstracts are highly acceptable. In the first experiment, an analysis of the variance (ANOVA) for text quality (Oakes 1998) showed differences among the three methods at $p \leq 0.005$ (observed $F(2, 27) = 9.66$). Tukey's multiple-comparison test (Byrkit 1987) shows statis-

Table 13
Results of human judgment in a categorization task and assessment about text quality.

Experiment	Summarization Methods					
First	Autosummarize		SumUM		Human	
15 evaluators	Success	Quality	Success	Quality	Success	Quality
10 documents	80%	1.46	80%	3.23	100%	4.25
Second	Autosummarize		SumUM		Human	
18 evaluators	Success	Quality	Success	Quality	Success	Quality
12 documents	70%	1.98	70%	3.15	80%	4.04
Third	n-STEIN		SumUM		Extractor	
20 evaluators	Success	Quality	Success	Quality	Success	Quality
15 documents	67%	2.76	80%	3.13	73%	3.47

tical differences in text quality at $p \leq 0.01$ for the two automatic systems (SumUM and Autosummarize), but no conclusion can be drawn about differences between the abstracts produced by those systems and the author abstract at levels 0.01 or 0.05. In the second experiment, the ANOVA showed differences at $p \leq 0.01$ between the three methods (observed $F(2, 33) = 10.35$). Tukey's test shows statistical differences at $p \leq 0.01$ between the two automatic systems (SumUM and Autosummarize) and differences with the author abstract at 0.05. In the third experiment, the ANOVA for text quality did not allow us to draw any conclusions about differences in text quality ($F(2, 42) = 0.83$).

5.2 Evaluation of Content in a Coselection Experiment

Our objective in the evaluation of content in a coselection experiment is to measure coselection between sentences selected by our system and a set of "correct" extracted sentences. This method of evaluation has already been used in other summarization evaluations such as Edmundson (1969) and Marcu (1997). The idea is that if we find a high degree of overlap between the sentences selected by an automatic method and the sentences selected by a human, the method can be regarded as effective. Nevertheless, this method of evaluation has been criticized not only because of the low rate of agreement between human subjects in this task (Jing et al. 1998), but also because there is no unique ideal or target abstract for a given document. Instead, there is a set of main ideas that a good abstract should contain (Johnson 1995). In our coselection experiment, we were also interested in comparing our system with other summarization technologies.

5.2.1 Materials.

Data used.   We used 10 technical articles from two different sources: 5 from the journal *Rapid Prototyping* and 5 from the journal *Internet Research*. The documents were downloaded from the Emerald Electronic Library. The abstracts and lists of keywords provided with the documents were deleted before the documents were used in the evaluation.

Reference extracts.   We used 30 automatic abstracts (three for each article) and nine assessors with a background in dealing with technical articles, on whom we relied to obtain an assessment of important sentences in the source documents. Eight assessors read two articles each, and one read four articles, because no other participants were available when the experiment was conducted. The assessor of each article chose a number of important sentences from that article (up to a maximum of N_i, the number of sentences chosen by the summarization methods). Each article was read by two different assessors; we thus had two sets of sentences for each article. We call these sets $S_{i,j}$ ($i \in [1 \ldots 10] \wedge j \in [1 \ldots 2]$). Most of the assessors found the task quite complex. Agreement between human assessors was only 37%.

Automatic extracts.   We considered three automatic systems in this evaluation: SumUM, Autosummarize, and Extractor. We produced three abstracts for each document. First we produced an abstract using SumUM. We counted the number of sentences selected by SumUM in order to produce the indicative-informative abstract (we verified that the number of sentences selected by the system represented between 10% and 25% of source documents). Then, we produced two other automatic abstracts, one using Autosummarize and another using Extractor. We specified to each system that it should select the same number of sentences as SumUM selected.

5.2.2 Procedure. We measure coselection between sentences produced by each method and the sentences selected by the assessors, computing recall, precision, and *F*-score as in Firmin and Chrzanowski (1999). In order to obtain a clear picture, we borrowed the scoring methodology proposed by Salton et al. (1997), additionally considering the following situations:

- *Union scenario:* For each document we considered the union of the sentences selected by the two assessors ($S_{i,1} \cup S_{i,2}$) and computed recall, precision, and *F*-score for each method.

- *Intersection scenario:* For each document we considered the intersection of the sentences selected by the two assessors ($S_{i,1} \cap S_{i,2}$) and computed recall, precision, and *F*-score for each method.

- *Optimistic scenario:* For each document and method we considered the case in which the method performed the best (highest *F*-score) and computed recall, precision, and *F*-score.

- *Pessimistic scenario:* For each document and method we considered the case in which the method performed the worst (lowest *F*-score) and computed recall, precision, and *F*-score.

5.2.3 Results and Discussion. For each scenario we present the average information in Table 14 (Saggion and Lapalme [2000c] presented detailed results of this experiment). For the scenario in which we consider the 20 human abstracts, SumUM obtained the best *F*-score in 60% of the cases, Extractor in 25% of the cases, and Autosummarize in 15% of the cases. If we assume that the sentences selected by the human assessors represent the most important or interesting information in the documents, then we can conclude that on average, SumUM performed better than the other two summarization technologies, even if these results are not exceptional in individual cases. An ANOVA showed statistical differences in the *F*-score measure at $p \leq 0.01$ between the different automatic abstracts (observed $F(2, 57) = 5.28$). Tukey's tests showed differences between SumUM and the two other automatic methods at $p \leq 0.01$.

    Here, we have compared three different methods of producing abstracts that are domain independent. Nevertheless, whereas Autosummarize and Extractor are truly text independent, SumUM is genre dependent: It was designed for the technical article and takes advantage of this fact in order to produce abstracts. We think that this

Table 14
Coselection between sentences selected by human assessors and sentences selected by three automatic summarization methods in recall (R), precision (P) and F-score (F).

	SumUM			Autosummarize			Extractor		
	R	P	F	R	P	F	R	P	F
Average	.23	.20	.21	.14	.11	.12	.12	.18	.14
Union	.21	.31	.25	.16	.19	.17	.11	.26	.15
Intersection	.28	.09	.14	.13	.04	.06	.08	.04	.06
Optimistic	.26	.23	.25	.16	.14	.15	.14	.25	.18
Pessimistic	.19	.17	.18	.11	.08	.09	.08	.11	.09

is the reason for the better performance of SumUM in this evaluation. The results of this experiment are encouraging considering the limited capacities of the actual implementation. We expect to improve the results in future versions of SumUM. Additional evaluations of SumUM using sentence acceptability criteria and content-based measures of indicativeness have been presented in Saggion and Lapalme (2000b) and Saggion (2000).

6. Related Work on Summarization

As a human activity, the production of summaries is directly associated with the processes of language understanding and production: A source text is read and understood to recognize its content, which is then compiled in a concise text. In order to explain this process, several theories have been proposed and tested in text linguistics, cognitive science, and artificial intelligence, including macro structures (Kintsch and van Dijk 1975; van Dijk 1977), history grammars (Rumelhart 1975), plot units (Lehnert 1981), and concept/coherence relations (Alterman and Bookman 1990). Computers have been producing summaries since the original work of Luhn (1958). Since then several methods and theories have been applied, including the use of term frequency $*$ inverse document frequency ($TF * IDF$) measures, sentence position, and cue and title words (Luhn 1958; Edmundson 1969; Kupiec, Pedersen, and Chen 1995; Brandow, Mitze, and Rau 1995); partial understanding using conceptual structures (DeJong 1982; Tait 1982); bottom-up understanding, top-down parsing, and automatic linguistic acquisition (Rau, Jacobs, and Zernik 1989); recognition of thematic text structures (Hahn 1990); cohesive properties of texts (Benbrahim and Ahmad 1995; Barzilay and Elhadad 1997); and rhetorical structure theory (Ono, Sumita, and Miike 1994; Marcu 1997).

In the context of the scientific article, Rino and Scott (1996) have addressed the problem of coherent selection for text summarization, but they depend on the availability of a complex meaning representation, which in practice is difficult to obtain from the raw text. Instead, superficial analysis in scientific text summarization using lexical information was applied by Lehmam (1997) for the French language. Liddy (1991) produced one of the most complete descriptions of conceptual information for abstracts of empirical research. In our work, we concentrated instead on conceptual information that is common across domains. Liddy's model includes three typical levels of information. The most representative level, called the **prototypical structure**, includes the information categories subjects, purpose, conclusions, methods, references, and hypotheses. The other two levels are the **typical structure** and the **elaborated structure**, which include information less frequently found in abstracts of empirical research. To our knowledge Liddy's model has never been implemented; nevertheless, it could be used as a starting point for improving our flat-domain model. Relevant work in rhetorical classification for scientific articles, which is the first step toward the production of scientific abstracts, is due to Teufel and Moens (1998), who used statistical approaches borrowed from Kupiec, Pedersen, and Chen (1995).

Our method is close to concept-based abstracting (CBA) (Jones and Paice 1992; Paice and Jones 1993) but differs from this approach in several aspects. CBA is used to produce abstracts of technical articles in specific domains, for example, in the domain of agriculture. Semantic roles such as species, cultivar, high-level property, and low-level property are first identified by the manual analysis of a corpus, and then patterns are specified that account for stylistic regularities of expression of the semantic roles in texts. These patterns are used in an information extraction process that instantiates the semantic roles. Selective analysis, although genre dependent, was developed as domain independent and tested in different technical domains without the need to

adapt the conceptual model, the patterns, or the conceptual dictionary. In order to adapt CBA to new domains, the semantic roles representing the "key" information in the new domain need to be identified, and new templates and patterns need to be constructed (Oakes and Paice 2001). Although such adaptation is generally done manually, recent work has shown how to export CBA to new domains automatically (Oakes and Paice 1999). CBA uses a fixed canned template for summary generation, whereas our method allows greater stylistic variability because the main "content" of the summary generated is expressed in the words of the authors of the paper. Selective analysis is used to produce indicative-informative abstracts, whereas CBA is mainly used to produce indicative abstracts, though some informative content is included in the form of extracted sentences containing results and conclusions (Paice and Oakes 1999). Our method can be seen as an extension of CBA that allows for domain independence and informativeness. We believe that the indicative patterns we have designed are genre dependent, whereas the informative patterns are general and can be used in any domain. Our implementation of patterns for information extraction is similar to Black's (1990) implementation of Paice's (1981) indicative phrases method, but whereas Black scores sentences based on indicative phrases contained in the sentences, our method scores the information from the sentences based on term distribution.

Our work in sentence reformulation is different from **cut-and-paste** summarization (Jing and McKeown 2000) in many ways. Jing (2000) proposes a novel algorithm for sentence reduction that takes into account different sources of information to decide whether or not to remove a particular component from a sentence to be included in a summary. The decision is made based on (1) the relation of the component to its context, (2) the probability of deleting such a component (estimated from a corpus of reduced sentences), and (3) linguistic knowledge about the essentiality of the component in the syntactic structure. Sentence reduction is concerned only with the removal of sentence components, so it cannot explain transformations observed in our corpus and in summarization in general, such as the reexpression of domain concepts and verbs. We achieve sentence reduction through a process of information extraction that extracts verbs and arguments, sometimes considering only sentence fragments (for example, initial prepositional phrases, parenthetical expressions, and some adverbials are ignored for some templates). The process removes domain concepts, avoids unnecessary grammatical subjects, and generates coordinate structures, avoiding verb repetition. Whereas our algorithm is genre dependent, requiring only shallow parsing, Jing's algorithm is genre and domain independent and requires full syntactic parsing and disambiguation and extensive linguistic resources.

Regarding the fusion of information, we have concentrated only on the fusion of explicit topical information (document topic, section topic, and signaling structural and conceptual elements). Jing and McKeown (2000) have proposed a rule-based algorithm for sentence combination, but no results have been reported. Radev and McKeown (1998) have already addressed the issue of information fusion in the context of multidocument summarization in one specific domain (i.e., terrorism): The fusion of information is achieved through the implementation of summary operators that integrate the information of different templates from different documents referring to the same event. Although those operators are dependent on the specific task of multidocument summarization, and to some extent on the particular domain they deal with, it is interesting to observe that some of Radev and McKeown's ideas could be applied in order to improve our texts. For example, their "refinement" operator could be used to improve the descriptions of the entities of the indicative abstract. The entities from the indicative abstract could be refined with definitions or descriptions from the InfoDB in order to obtain a better and more compact text. The idea of

elaborating topics has also been addressed by Mani, Gates, and Bloedorn (1999). They have proposed a number of rules for summary revision aiming at conciseness; their elimination rule discards parenthetical and initial prepositional phrases, as does our approach. Their aggregation operation combines two constituents on the basis of referential identity and so is more general than our combination of topical information. Although their approach is domain independent, it requires full syntactic analysis and coreference resolution.

7. Conclusions

SumUM has been fully implemented to take a raw text as input and produce a summary. This involves the following successive steps: text segmentation, part-of-speech tagging, partial syntactic and semantic analysis, sentence classification, template instantiation, content selection, text regeneration, and topic elaboration. Our research was based on the intensive study of manual alignments between sentences of professional abstracts and elements of source documents and on the exploration of the essential differences between indicative and informative abstracts.

Although our method was deeply influenced by the results of our corpus study, it nevertheless has many points in common with recent theoretical and programmatic directions in automatic text summarization. For example, Sparck Jones (1997) argues in favor of a kind of "indicative, skeletal summary" and the need to explore dynamic, context-sensitive summarization in interactive situations in which the summary changes according to the user needs. Hutchins (1995) advocates indicative summaries, produced from parts of a document in which the topics are likely to be stated. These abstracts are well suited for situations in which the actual user is unknown (i.e., a general reader), since the abstract will provide the reader with good entry points for retrieving more detailed information. If the users are known, the abstract can be tailored to their specific profiles; such profiles might specify the reader's interest in various types of information, such as conclusions, definitions, methods, or user needs expressed in a "query" to an information retrieval system (Tombros, Sanderson, and Gray 1998). Our method, however, was designed without any particular reader in mind and with the assumption that a text does have a "main" topic.

In this article, we have presented an evaluation of automatically generated indicative-informative abstracts in terms of content and text quality. In the evaluation of the indicative content in a categorization task, no differences were observed among the different automatic systems. The automatic abstracts generated by SumUM were considered more acceptable than other systems' abstracts. In the evaluation of the informative content, SumUM selected sentences that were evaluated as more relevant by human assessors than sentences selected by other summarization technologies; statistical tests showed significant differences between the automatic methods in that evaluation. In the future, we plan to address several issues, including the study of robust automatic text classification techniques, anaphora resolution, and lexical cohesion for improving elaboration of topics as well as the incorporation of local discourse analysis to improve the coherence of abstracts.

References

Alterman, Richard and
    Lawrence A. Bookman. 1990. Some
    computational experiments in
    summarization. *Discourse Processes*,
    13:143–174.
American National Standards Institute.
    (ANSI). 1979. *Writing Abstracts*.
Barzilay, Regina and Michael Elhadad. 1997.
    Using lexical chains for text
    summarization. In *Proceedings of the
    ACL/EACL'97 Workshop on Intelligent
    Scalable Text Summarization*, pages 10–17,
    Madrid, July.
Benbrahim, Mohamed and Kurshid Ahmad.
    1995. Text summarisation: The role of
    lexical cohesion analysis. *New Review of
    Document & Text Management*, 1:321–335.
Bernier, Charles L. 1985. Abstracts and
    abstracting. In E. D. Dym, editor, *Subject
    and Information Analysis*, volume 47 of
    Books in Library and Information Science.
    Marcel Dekker, Inc., pages 423–444.
Bhatia, Vijay K. 1993. *Analysing Genre:
    Language Use in Professional Settings*.
    Longman.
Black, William J. 1990. Knowledge based
    abstracting. *Online Review*, 14(5):327–340.
Brandow, Ronald, K. Mitze, and Lisa F. Rau.
    1995. Automatic condensation of
    electronic publications by sentence
    selection. *Information Processing &
    Management*, 31(5):675–685.
Bunge, Mario. 1967. *Scientific Research I. The
    Search for System.* Springer-Verlag, New
    York.
Byrkit, Donald R. 1987. *Statistics Today: A
    Comprehensive Introduction*.
    Benjamin/Cummings.

Cremmins, Eduard T. 1982. *The Art of
    Abstracting*. ISI Press.
DeJong, Gerald. 1982. An overview of the
    FRUMP system. In W. G. Lehnert and
    M. H. Ringle, editors, *Strategies for Natural
    Language Processing*. Lawrence Erlbaum,
    pages 149–176.
Edmundson, H. P. 1969. New methods in
    automatic extracting. *Journal of the
    Association for Computing Machinery*,
    16(2):264–285.
Educational Resources Information Center
    (ERIC). 1980. *Processing Manual: Rules and
    Guidelines for the Acquisition, Selection, and
    Technical Processing of Documents and
    Journal Articles by the Various Components of
    the ERIC Network*. ERIC.
Fellbaum, Christiane, editor. 1998. *WordNet:
    An Electronic Lexical Database*. MIT Press.
Filman, Roger E. and Sangan Pant. 1998.
    Searching the Internet. *IEEE Internet
    Computing*, 2(4):21–23.
Firmin, Thérèse and
    Michael J. Chrzanowski. 1999. An
    evaluation of automatic text
    summarization systems. In I. Mani and
    M. T. Maybury, editors, *Advances in
    Automatic Text Summarization*. MIT Press,
    pages 325–336.
Foster, George. 1991. Statistical lexical
    disambiguation. Master's thesis, School of
    Computer Science, McGill University,
    Montréal, Québec, Canada.
Gibson, Timothy R. 1993. *Towards a Discourse
    Theory of Abstracts and Abstracting*.
    Department of English Studies,
    University of Nottingham.
Grant, Pamela. 1992. *The Integration of Theory
    and Practice in the Development of
    Summary-Writing Strategies*. Ph.D. thesis,
    Faculté des Études Supérieures,
    Université de Montréal.
Hahn, Udo. 1990. Topic parsing: Accounting
    for text macro structures in full-text
    analysis. *Information Processing &
    Management*, 26(1):135–170.
Hutchins, John. 1995. Introduction to text
    summarization workshop. In
    B. Engres-Niggemeyer, J. Hobbs, and
    K. Sparck Jones, editors, *Summarising Text
    for Intelligent Communication*, Dagstuhl
    Seminar Report 79. IBFI, Schloss
    Dagstuhl, Wadern, Germany,
INSPEC. 2000. *INSPEC database for physics,
    electronics and computing*.
    http://www.iee.org.uk/publish/inspec/
Jing, Hongyan. 2000. Sentence reduction for
    automatic text summarization. In
    *Proceedings of the Sixth Applied Natural
    Language Processing Conference*, pages
    310–315, Seattle, April 29–May 4.

Jing, Hongyan and Kathleen McKeown. 2000. Cut and paste based text summarization. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 178–185, Seattle, April 29–May 4.

Jing, Hongyan, Kathleen McKeown, Regina Barzilay, and Michael Elhadad. 1998. Summarization evaluation methods: Experiments and analysis. In *Intelligent Text Summarization: Papers from the 1998 AAAI Spring Symposium*. Stanford, March 23–25. Technical Report SS-98-06, AAAI Press, pages 60–68.

Johnson, Frances. 1995. Automatic abstracting research. *Library Review*, 44(8):28–36.

Johnson, Frances C., Chris D. Paice, William J. Black, and A. P. Neal. 1993. The application of linguistic processing to automatic abstract generation. *Journal of Document & Text Management*, 1(3):215–241.

Jones, Paul A. and Chris D. Paice. 1992. A "select and generate" approach to automatic abstracting. In A. M. McEnry and C. D. Paice, editors, *Proceedings of the 14th British Computer Society Information Retrieval Colloquium*. Springer Verlag, pages 151–154.

Jordan, Michael P. 1993. Openings in very formal technical texts. *Technostyle*, 11(1):1–26.

Jordan, Michael P. 1996. *The Language of Technical Communication: A Practical Guide for Engineers, Technologists and Technicians*. Quarry.

Kaplan, Robert B., Selena Cantor, Cynthia Hagstrom, Lia D. Kamhi-Stein, Yumiko Shiotani, and Cheryl B. Zimmerman. 1994. On abstract writing. *Text*, 14(3):401–426.

Kintsch, Walter and Teun A. van Dijk. 1975. Comment on se rappelle et on résume des histoires. *Langages*, 40:98–116.

Knight, Kevin and Daniel Marcu. 2000. Statistics-based summarization—Step one: Sentence compression. In *Proceedings of the 17th National Conference of the American Association for Artificial Intelligence (AAAI)*, July 30–August 3.

Kupiec, Julian, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th ACM-SIGIR Conference*, pages 68–73.

Lehmam, Abderrafih. 1997. Une structuration de texte conduisant à la construction d'un système de résumé automatique. In *Actes des journées scientifiques et techniques du réseau francophone de l'ingénierie de la langue de l'AUPELF-UREF*, pages 175–182, 15–16 April.

Lehnert, Wendy. 1981. Plot units and narrative summarization. *Cognitive Science*, 5:293–331.

Liddy, Elizabeth D. 1991. The discourse-level structure of empirical abstracts: An exploratory study. *Information Processing & Management*, 27(1):55–81.

Luhn, Hans P. 1958. The automatic creation of literature abstracts. *IBM Journal of Research Development*, 2(2):159–165.

Maizell, Robert E., Julian F. Smith, and Tibor E. R. Singer. 1971. *Abstracting Scientific and Technical Literature*. Wiley-Interscience.

Mani, Inderjeet, Barbara Gates, and Eric Bloedorn. 1999. Improving summaries by revising them. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 558–565, College Park, MD, 20–26 June.

Mani, Inderjeet, David House, Gary Klein, Lynette Hirshman, Leo Obrst, Thérèse Firmin, Michael Chrzanowski, and Beth Sundheim. 1998. The TIPSTER SUMMAC text summarization evaluation. Technical Report, Mitre Corporation.

Marcu, Daniel. 1997. From discourse structures to text summaries. In *Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*, pages 82–88, Madrid, July 11.

Mariani, Joseph. 1995. Evaluation. In Ronald E. Cole, editor, *Survey of the State of the Art in Human Language Technology*. Cambridge University Press, chapter 13, pages 475–518.

Mathis, Betty A. and James E. Rush. 1985. Abstracting. In E. D. Dym, editor, *Subject and Information Analysis*, volume 47 of *Books in Library and Information Science*. Marcel Dekker, pages 445–484.

n-STEIN. 2000. n-STEIN Web page. http://www.gespro.com.

Oakes, Michael P. 1998. *Statistics for Corpus Linguistics*. Edinburgh University Press.

Oakes, Michael P. and Chris D. Paice. 1999. The automatic generation of templates for automatic abstracting. In *21st BCS IRSG Colloquium on IR*, Glasgow.

Oakes, Michael P. and Chris D. Paice. 2001. Term extraction for automatic abstracting. In D. Bourigault, C. Jacquemin, and M.-C. L'Homme, editors, *Recent Advances in Computational Terminology*, volume 2 of *Natural Language Processing*. John Benjamins, chapter 17, pages 353–370.

Ono, Kenji, Kazuo Sumita, and Seiji Miike. 1994. Abstract generation based on rhetorical structure extraction. In *Proceedings of the International Conference on*

Computational Linguistics, pages 344–348.

Paice, Chris D. 1981. The automatic generation of literary abtracts: An approach based on identification of self-indicating phrases. In O. R. Norman, S. E. Robertson, C. J. van Rijsbergen, and P. W. Williams, editors, *Information Retrieval Research*. London: Butterworth, pages 172–191.

Paice, Chris D. 1991. The rhetorical structure of expository text. In K. P. Jones, editor, *Informatics 11: The Structuring of Information*, University of York, 20–22 March. Aslib.

Paice, Chris D., William J. Black, Frances C. Johnson, and A. P. Neal. 1994. Automatic abstracting. R&D Report 6166, British Library.

Paice, Chris D. and Paul A. Jones. 1993. The identification of important concepts in highly structured technical papers. In R. Korfhage, E. Rasmussen, and P. Willett, editors, *Proceedings of the 16th ACM-SIGIR Conference*, pages 69–78.

Paice, Chris D. and Michael P. Oakes. 1999. A concept-based method for automatic abstracting. Research Report 27, Library and Information Commission.

Radev, Dragomir R. and Kathleen R. McKeown. 1998. Generating natural language summaries from multiple on-line sources. *Computational Linguistics*, 24(3):469–500.

Rau, Lisa F., Paul S. Jacobs, and Uri Zernik. 1989. Information extraction and text summarization using linguistic knowledge acquisition. *Information Processing & Management*, 25(4):419–428.

Rino, Lucia H. M. and Donia Scott. 1996. A discourse model for gist preservation. In D. L. Borges and C. A. A. Kaestner, editors, *Proceedings of the 13th Brazilian Symposium on Artificial Intelligence (SBIA'96): Advances in Artificial Intelligence*, October 23–25, Curitiba, Brazil. Springer, pages 131–140.

Rowley, Jennifer. 1982. *Abstracting and Indexing*. Clive Bingley, London.

Rumelhart, David E. 1975. Notes on a schema for stories. In *Language, Thought, and Culture: Advances in the Study of Cognition*. Academic Press.

Saggion, Horacio. 2000. *Génération automatique de résumés par analyse sélective*. Ph.D. thesis, Département d'informatique et de recherche opérationnelle, Faculté des arts et des sciences, Université de Montréal, Montréal.

Saggion, Horacio and Guy Lapalme. 2000a. Concept identification and presentation in the context of technical text summarization. In *Proceedings of the Workshop on Automatic Summarization (ANLP-NAACL2000)*, Seattle, 30 April. Association for Computational Linguistics.

Saggion, Horacio and Guy Lapalme. 2000b. Selective analysis for automatic abstracting: Evaluating indicativeness and acceptability. In *Proceedings of the Computer-Assisted Information Searching on Internet Conference (RIAO'2000)*, Paris, 12–14 April.

Saggion, Horacio and Guy Lapalme. 2000c. Summary generation and evaluation in SumUM. In *Advances in Artificial Intelligence. International Joint Conference: Seventh Ibero-American Conference on Artificial Intelligence and 15th Brazilian Symposium on Artificial Intelligence (IBERAMIA-SBIA 2000)*, volume 1952 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, pages 329–338.

Salton, Gerald, Amit Singhal, Mandar Mitra, and Chris Buckley. 1997. Automatic text structuring and summarization. *Information Processing & Management*, 33(2):193–207.

Sharp, Bernadette. 1998. *Elaboration and Testing of New Methodologies for Automatic Abstracting*. Ph.D. thesis, University of Aston in Birmingham.

SICStus. 1998. *SICStus Prolog User's Manual*. SICStus.

Sparck Jones, Karen. 1993. What might be in a summary? In K. Knorz and C. Womser-Hacker, editors, *Information Retrieval 93: Von der Modellierung zur Anwendung*.

Sparck Jones, Karen. 1997. Document processing: Summarization. In R. Cole, editor, *Survey of the State of the Art in Human Language Technology*. Cambridge University Press, chapter 7, pages 266–269.

Sparck Jones, Karen and Brigitte Endres-Niggemeyer. 1995. Automatic summarizing. *Information Processing & Management*, 31(5):625–630.

Sparck Jones, Karen and Julia R. Galliers. 1995. *Evaluating Natural Language Processing Systems: An Analysis and Review*, number 1083 in *Lecture Notes in Artificial Intelligence*. Springer.

Tait, John I. 1982. *Automatic Summarising of English Texts*. Ph.D. thesis, Computer Laboratory, Cambridge University, Cambridge.

Teufel, S. 1998. Meta-discourse markers and problem-structuring in scientific texts. In M. Stede, L. Wanner, and E. Hovy, editors, *Proceedings of the Workshop on*

Discourse Relations and Discourse Markers (COLING-ACL'98), pages 43–49, 15 August.

Teufel, S. and M. Moens. 1998. Sentence extraction and rhetorical classification for flexible abstracts. In *Intelligent Text Summarization: Papers from the 1998 AAAI Spring Symposium*, Stanford, March 23–25. Technical Report SS-98-06, AAAI Press, pages 16–25.

Tombros, Anastasios, Mark Sanderson, and Phil Gray. 1998. Advantages of query biased summaries in information retrieval. In *Intelligent Text Summarization: Papers from the 1998 AAAI Spring Symposium*, Stanford, March 23–25. Technical Report SS-98-06, AAAI Press, pages 34–43.

Turney, Peter D. 1999. Learning to extract keyphrases from text. Technical Report ERB-1051, National Research Council of Canada.

van Dijk, Teun A. 1977. Recalling and summarizing complex discourse. In W. Burghardt and K. Holzer, editors, *Text Processing*. De Gruyter, New York and Berlin, pages 49–118.

Vianna, Fernando de Melo, editor. 1980. *Roget's II: The New Thesaurus*. Houghton Mifflin, Boston.

Wall, Larry, Tom Christiansen, and Randal L. Schwartz. 1996. *Programming Perl*. O'Reilly & Associates, second edition.